Date=25/09/2020
Lecture By=Arkesh Jaiswal
Subject ⇒ CSS

IN PREVIOUS LECTURE **(QUICK RECAP) Date-24/09/2020**	**In Today's Lecture (Overview)**
Tag CSS margin Property CSS background-attachment Property CSS object-position Property CSS padding Property MCQS Questions for self practice / CC For the day	CSS Rounded Corners CSS Gradients CSS Shadow Effects CSS 3D Transforms CSS Transitions Questions For Self Practice / Assignment and CC for the Day

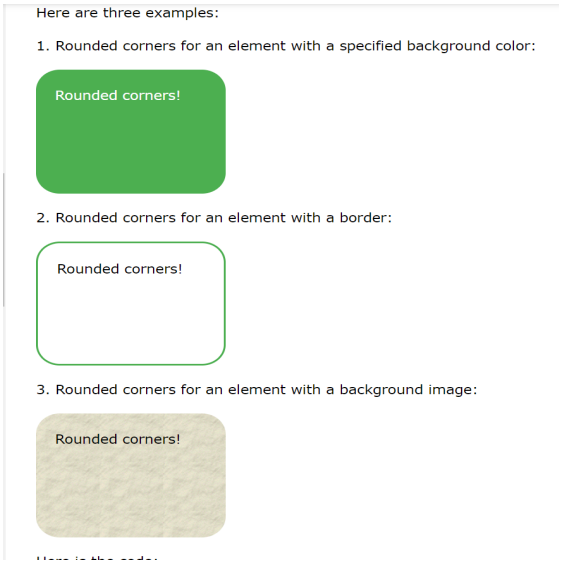
CSS Rounded Corners

With the CSS `border-radius` property, you can give any element "rounded corners".

CSS border-radius Property

The CSS `border-radius` property defines the radius of an element's corners.

This property allows you to add rounded corners to elements!



Here is the code:

Example

```
#rcorners1 {
  border-radius: 25px;
  background: #73AD21;
  padding: 20px;
  width: 200px;
  height: 150px;
}

#rcorners2 {
  border-radius: 25px;
  border: 2px solid #73AD21;
```

```
  padding: 20px;
  width: 200px;
  height: 150px;
}
#rcorners3 {
  border-radius: 25px;
  background: url(paper.gif);
  background-position: left top;
  background-repeat: repeat;
  padding: 20px;
  width: 200px;
  height: 150px;
}
```

CSS border-radius - Specify Each Corner

The `border-radius` property can have from one to four values. Here are the rules:

- Four values - border-radius: 15px 50px 30px 5px; (first value applies to top-left corner, second value applies to top-right corner, third value applies to bottom-right corner, and fourth value applies to bottom-left corner):

- Three values - border-radius: 15px 50px 30px; (first value applies to top-left corner, second value applies to top-right and bottom-left corners, and third value applies to bottom-right corner):
- Two values - border-radius: 15px 50px; (first value applies to top-left and bottom-right corners, and the second value applies to top-right and bottom-left corners):
- One value - border-radius: 15px; (the value applies to all four corners, which are rounded equally:

CSS Rounded Corners Properties

Property	Description
border-radius	A shorthand property for setting all the four border-*-*-radius properties
border-top-left-radius	Defines the shape of the border of the top-left corner
border-top-right-radius	Defines the shape of the border of the top-right corner
border-bottom-right-radius	Defines the shape of the border of the bottom-right corner
border-bottom-left-radius	Defines the shape of the border of the bottom-left corner

CSS Gradients

CSS gradients let you display smooth transitions between two or more specified colors.

CSS defines two types of gradients:

- Linear Gradients (goes down/up/left/right/diagonally)
- Radial Gradients (defined by their center)

CSS Linear Gradients

To create a linear gradient you must define at least two color stops. Color stops are the colors you want to render smooth transitions among. You can also set a starting point and a direction (or an angle) along with the gradient effect.

Syntax

```
background-image: linear-gradient(direction, color-stop1,
color-stop2, ...);
```

Direction - Top to Bottom (this is default)

The following example shows a linear gradient that starts at the top. It starts red, transitioning to yellow:

top to bottom (default)

Example

```
#grad { background-image: linear-gradient(red, yellow);}
```

Direction - Left to Right

The following example shows a linear gradient that starts from the left. It starts red, transitioning to yellow:

Example

```
#grad {background-image: linear-gradient(to right, red , yellow)}
```

Direction - Diagonal

You can make a gradient diagonally by specifying both the horizontal and vertical starting positions.

The following example shows a linear gradient that starts at top left (and goes to bottom right). It starts red, transitioning to yellow:

Example

```
#grad {background-image: linear-gradient(to bottom right, red, yellow);}
```

CSS Shadow Effects

With CSS you can add shadow to text and to elements.

In these chapters you will learn about the following properties:

- `text-shadow`
- `box-shadow`

CSS Text Shadow

The CSS `text-shadow` property applies shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

Example

```
h1 {text-shadow: 2px 2px;}
```

Example

```
h1 {text-shadow: 2px 2px red;}
```

Then, add a blur effect to the shadow:

```
h1 {text-shadow: 2px 2px 5px red;}
```

The following example shows a white text with black shadow:

Example

```
h1 {color: white;

text-shadow: 2px 2px 4px #000000;}
```

The following example shows a red neon glow shadow:

Example

```
h1 {text-shadow: 0 0 3px #FF0000;}
```

Multiple Shadows

To add more than one shadow to the text, you can add a comma-separated list of shadows.

The following example shows a red and blue neon glow shadow:

Example

```
h1 {text-shadow: 0 0 3px #FF0000, 0 0 5px #0000FF;}
```

The following example shows a white text with black, blue, and darkblue shadow

Example

```
h1 {color: white;

 text-shadow: 1px 1px 2px black, 0 0 25px blue, 0 0 5px darkblue;}
```

You can also use the text-shadow property to create a plain border around some text (without shadows):

Example

```
h1 {color: yellow;

  text-shadow: -1px 0 black, 0 1px black, 1px 0 black, 0 -1px black;}
```

CSS 3D Transforms

CSS also supports 3D transformations.

Mouse over the elements below to see the difference between a 2D and a 3D transformation

CSS 3D Transforms Methods

With the CSS `transform` property you can use the following 3D transformation methods:

- `rotateX()`
- `rotateY()`
- `rotateZ()`

CSS 3D Transform Methods

Function	Description
matrix3d (*n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n*)	Defines a 3D transformation, using a 4x4 matrix of 16 values
translate3d(*x,y,z*)	Defines a 3D translation
translateX(*x*)	Defines a 3D translation, using only the value for the X-axis
translateY(*y*)	Defines a 3D translation, using only the value for the Y-axis
translateZ(*z*)	Defines a 3D translation, using only the value for the Z-axis
scale3d(*x,y,z*)	Defines a 3D scale transformation
scaleX(*x*)	Defines a 3D scale transformation by giving a value for the X-axis

scaleY(*y*)	Defines a 3D scale transformation by giving a value for the Y-axis
scaleZ(*z*)	Defines a 3D scale transformation by giving a value for the Z-axis
rotate3d(*x,y,z,angle*)	Defines a 3D rotation
rotateX(*angle*)	Defines a 3D rotation along the X-axis
rotateY(*angle*)	Defines a 3D rotation along the Y-axis
rotateZ(*angle*)	Defines a 3D rotation along the Z-axis
perspective(*n*)	Defines a perspective view for a 3D transformed element

CSS Transitions

CSS transitions allows you to change property values smoothly, over a given duration.

Mouse over the element below to see a CSS transition effect:

Here you will learn about the following properties:

- `transition`
- `transition-delay`
- `transition-duration`
- `transition-property`
- `transition-timing-function`

How to Use CSS Transitions?

To create a transition effect, you must specify two things:

- the CSS property you want to add an effect to
- the duration of the effect

Note: If the duration part is not specified, the transition will have no effect, because the default value is 0.

The following example shows a 100px * 100px red element. The element has also specified a transition effect for the width property, with a duration of 2 seconds:

Specify the Speed Curve of the Transition

The `transition-timing-function` property specifies the speed curve of the transition effect.

The transition-timing-function property can have the following values:

- `ease` - specifies a transition effect with a slow start, then fast, then end slowly (this is default)
- `linear` - specifies a transition effect with the same speed from start to end

- `ease-in` - specifies a transition effect with a slow start
- `ease-out` - specifies a transition effect with a slow end
- `ease-in-out` - specifies a transition effect with a slow start and end
- `cubic-bezier(n,n,n,n)` - lets you define your own values in a cubic-bezier function

The following example shows the some of the different speed curves that can be used:

Example

```
#div1 {transition-timing-function: linear;}
#div2 {transition-timing-function: ease;}
#div3 {transition-timing-function: ease-in;}
#div4 {transition-timing-function: ease-out;}
#div5 {transition-timing-function: ease-in-out;}
```

The following table lists all the CSS transition properties:

Property	Description
transition	A shorthand property for setting the four transition properties into a single property
transition-delay	Specifies a delay (in seconds) for the transition effect
transition-duration	Specifies how many seconds or milliseconds a transition effect takes to complete
transition-property	Specifies the name of the CSS property the transition effect is for

Questions For Self Practice / Assignment and CC for the Day

1.Use the company website built previously and add to it a fixed navbar a footer which
changes colour when hovered.

2.Animate the background and size of a button when hovered